Assignment 3: Modeling assignment: Using meta learning schemes with a stong and a weak learner for classification

By: Rylee Texter

The first model we are going to create is a cost sensitive classifier combine with bagging and J48 learner.

Cost Sensitive classifier combined with bagging and J48 Learner

In this model, I changed the type 2 error from 1 to 5, going to increments of 1. I found the model with the best metrics to be when the type 2 error was set to 4. Here are the details of that model:

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0.894	0.151	0.656	0.894	0.757	0.677	0.896	0.686	fp
	0.849	0.106	0.961	0.849	0.902	0.677	0.896	0.947	nfp
Weighted Avg.	0.860	0.117	0.887	0.860	0.866	0.677	0.896	0.883	

=== Confusion Matrix ===

```
  a   b   <-- classified as
 42   5 |   a = fp
 22 124 |   b = nfp
```

Important Numbers:

Precision: **0.887**
Type 1 error rate: 22/(22+124) = **15.1%**
Type 2 error rate: 5/(5+42) = **10.6%**

Now, in order to verify this model, we are going to use our test data set:

Cost Sensitive Classifier combined with bagging and J48 Learner Using Test Data Set

Remember, we are still keeping the type 2 error set at 4 because that is what had the lowest error rates in general and the highest precision. Here are the details of this model:

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0.833	0.151	0.645	0.833	0.727	0.632	0.887	0.754	fp
	0.849	0.167	0.939	0.849	0.892	0.632	0.887	0.945	nfp
Weighted Avg.	0.845	0.163	0.867	0.845	0.851	0.632	0.887	0.898	

=== Confusion Matrix ===

a	b	<-- classified as
20	4	a = fp
11	62	b = nfp

Important Numbers:
Precision: 0.867
Type 1 error rate: 15.1%
Type 2 error rate: 16.7%

Because the error rates are still fairly low and the precision is still high, this model is verified. Therefore, for the cost sensitive classifier combined with bagging and J48 learner, type 2 error set to approximately 4 will obtain the best model in terms of the highest precision and the lowest error rates. Though, it is important to note that the type 1 error rate did not increase while the type 2 error did increase. Type 2 error is more detrimental than type 1, so there is still room for improvement here.

Cost Sensitive Classifier Combined with Bagging and Decision Stump

For the next model, we are going to create a cost sensitive classifier combined with bagging and decision stump learners. Furthermore, we are going to change the Type 2 error in order to find the optimal model.

After changing the Type 2 error from 1 to 3 with increments of 0.5, I found that the optimal model was the one with the type 2 error set to 2. Here are the details of this model:

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0.872	0.151	0.651	0.872	0.745	0.661	0.892	0.691	fp
	0.849	0.128	0.954	0.849	0.899	0.661	0.892	0.943	nfp
Weighted Avg.	0.855	0.133	0.880	0.855	0.861	0.661	0.892	0.881	

=== Confusion Matrix ===

```
  a   b   <-- classified as
 41   6 |   a = fp
 22 124 |   b = nfp
```

Important Numbers:
Precision: 0.88
Type 1 error rate: 15.1%
Type 2 error rate: 12.8%

In order to verify this model, we are now going to test it using the test data set. Remember the type 2 error will continue to be set to 2. Here are the details of this model:

Cost sensitive Classifier combined with bagging and Decision Stump Using Test Data Set

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0.875	0.233	0.553	0.875	0.677	0.568	0.876	0.641	fp
	0.767	0.125	0.949	0.767	0.848	0.568	0.876	0.935	nfp

```
Weighted Avg.    0.794    0.152    0.851     0.794    0.806     0.568    0.876     0.862

=== Confusion Matrix ===

  a  b   <-- classified as
 21  3 |  a = fp
 17 56 |  b = nfp
```

Important Numbers:
Precision: 0.851
Type 1 error rate: 23.3%
Type 2 error rate: 12.5%

This model still shows a high precision, which is very good. The type 1 error rate increased by an amount of 8.2%, which is not ideal at first glance. But, the type 2 error actually decreased by 0.3% which is important. Because type 2 is more serious than type 1, this is actually a positive aspect of this model.

Cost Sensitive classifier combined with boosting (AdaBoostM1) and J48

Now, we aim to create a cost sensitive classifier combined with boosting and J48. Again, we will change the type 2 error to find the optimal model.

I changed the type 2 error from 1 to 5 with increments of 1, and found that one of the optimal models is found when the type 2 error is set to 2. Here are the details of this model:

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0.787	0.082	0.755	0.787	0.771	0.695	0.905	0.749	fp
	0.918	0.213	0.931	0.918	0.924	0.695	0.905	0.951	nfp
Weighted Avg.	0.886	0.181	0.888	0.886	0.887	0.695	0.905	0.902	

=== Confusion Matrix ===

```
  a   b   <-- classified as
 37  10 |   a = fp
 12 134 |   b = nfp
```

Important Numbers:
Precision: 0.888
Type 1 error rate: 8.2%
Type 2 error rate: 21.3 %

This model has a high type 2 error rate. Even though this was one of the best models found, there is still great room for improvement on this model.

Now, we are going to verify the first model, where type 2 error is set to 2.

Cost Sensitive Classifier combine with boosting (AdaBoostM1) and J48 using test set

Here are the details of this model:

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall   F-Measure MCC   ROC Area PRC Area
Class

	0.625	0.123	0.625	0.625	0.625	0.502	0.879	0.702	fp
	0.877	0.375	0.877	0.877	0.877	0.502	0.879	0.946	nfp
Weighted Avg.	0.814	0.313	0.814	0.814	0.814	0.502	0.879	0.885	

=== Confusion Matrix ===

```
  a  b   <-- classified as
 15  9 |  a = fp
  9 64 |  b = nfp
```

Important Numbers:
Precision: 0.814
Type 1 error rate: 12.3%
Type 2 error rate: 37.5%

As we can see, there was a high jump in the type 2 error rate. This is not good. Once I tested this model changing type 2 errors further, I found that when the type 2 error is set to 4, there is the lowest type 2 error rate between the cross validation and the test set. Here are the details of this model:

Type 2 error set to 4
Cross Validation:

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0.702	0.089	0.717	0.702	0.710	0.618	0.892	0.711	fp
	0.911	0.298	0.905	0.911	0.908	0.618	0.892	0.948	nfp
Weighted Avg.	0.860	0.247	0.859	0.860	0.860	0.618	0.892	0.890	

=== Confusion Matrix ===

```
   a   b   <-- classified as
  33  14 |   a = fp
  13 133 |   b = nfp
```

Using Test Set:

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area
Class
0.667 0.123 0.640 0.667 0.653 0.536 0.842 0.640 fp
0.877 0.333 0.889 0.877 0.883 0.536 0.842 0.924 nfp
Weighted Avg. 0.825 0.281 0.827 0.825 0.826 0.536 0.842 0.854

=== Confusion Matrix ===

a b <-- classified as
16 8 | a = fp
9 64 | b = nfp

As we can see, the test set model has a lower type 2 error rate of 33.3% compared to the other model (type 2 error set to 2) of 37.5. Both of these error rates are still high, and I think it is important to consider other options and how to further this model before continuing or deciding between the two.

Therefore, I found two optimal models for this section. One where the type 2 error is set to 2, and another where the type 2 error is set to 4.

Cost Sensitive Classifier Combined with boosting (AdaBoostM1) and Decision Stump

Now in this section, we aim to create a closs sensitive classifier, combing with boosting (AdaBoostM1) and Decision Stump learners. We aim to change the type 2 error to find the optimal model.

After changing the type 2 error from 1 to 5 with increments of one, I found that the optimal model was found when type 2 error was set to 3. Here are details on this model with 10 folds cross validation:

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area
Class
0.915 0.178 0.623 0.915 0.741 0.660 0.880 0.690 fp
0.822 0.085 0.968 0.822 0.889 0.660 0.880 0.942 nfp

Weighted Avg.	0.845	0.108	0.884	0.845	0.853	0.660	0.880	0.881

=== Confusion Matrix ===

```
  a   b   <-- classified as
 43   4 |   a = fp
 26 120 |   b = nfp
```

Important Numbers:
Precision: 0.884
Type 1 error rate: 17.8%
Type 2 error rate: 8.5%

This is a very high precision and a better model. Though the type 1 error rate is a little on the higher side, the type 2 error rate is low at 8.5% which is more important. Now we are going to validate this model with a test set.

Cost Sensitive Classifier Combined with Boosting and Decision Stump using Test Set

Here are the details of this model:

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0.875	0.233	0.553	0.875	0.677	0.568	0.874	0.655	fp
	0.767	0.125	0.949	0.767	0.848	0.568	0.874	0.931	nfp
Weighted Avg.	0.794	0.152	0.851	0.794	0.806	0.568	0.874	0.863	

=== Confusion Matrix ===

```
  a  b   <-- classified as
 21  3 |  a = fp
 17 56 |  b = nfp
```

Important Numbers:
Precision: 0.851
Type 1 error rate: 23.3%
Type 2 error rate: 12.5%

Though the model did not perform as well as the cross validation model, it still maintained a high precision of 0.851 which is very good. In addition, the type 1 and 2 error rates raised a little, but the type 2 error rate is fairly low at 12.5%. It is important to further investigate.

Setting the number of iterations of each meta learner to 25

Now, we are going to repeat the processes above, creating four models, but this time each meta learner will have 25 iterations. We will continue to find the optimal type 2 error for the model, and we will reevaluate and draw conclusions on each model. Then after we've created and evaluated the models, we will compare models with the regular 10 iterations, and the new models with the 25 iterations.

Cost Sensitive Classifier Combined with bagging and J48

Now we are going to recreate this model with 10 fold cross validation, but the iterations are going to be set to 25. We will then change the type 2 error to find the optimal model.

I found the optimal model to be when the type 2 error was set to 4. Here are the details of this model:

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0.915	0.144	0.672	0.915	0.775	0.703	0.904	0.687	fp
	0.856	0.085	0.969	0.856	0.909	0.703	0.904	0.952	nfp
Weighted Avg.	0.870	0.099	0.897	0.870	0.876	0.703	0.904	0.888	

=== Confusion Matrix ===

```
  a   b   <-- classified as
 43   4 |   a = fp
 21 125 |   b = nfp
```

Important Numbers:
Precision: 0.897
Type 1 error rate: 14.4%
Type 2 error rate: 8.5%

This model has a high precision of almost 90%. This is a very good model. Furthermore, the type 2 error is below 10% which is also good. Even though the type 1 error rate is about 10%, because it is a less costly error, it is okay.

Now we are going to validate this model using a test set. Remember the iterations continue to stay at 25.

Cost Sensitive Classifier Combined with Bagging and J48 Using Test Set

Here are the details of this model:

```
=== Detailed Accuracy By Class ===

              TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area
Class
              0.792    0.164    0.613      0.792    0.691      0.580    0.880     0.727     fp
              0.836    0.208    0.924      0.836    0.878      0.580    0.880     0.948     nfp
Weighted Avg.    0.825    0.197    0.847      0.825    0.831      0.580    0.880     0.893

=== Confusion Matrix ===

  a  b   <-- classified as
 19  5 |  a = fp
 12 61 |  b = nfp
```

Important Numbers:
Precision: 0.847
Type 1 error rate: 16.4%
Type 2 error rate: 20.8%

Overall, this model continues to have a higher precision when validated on a test set. Though, the type 2 error jumped from 8.5% to 20.8%, which is not very good and it is important to have further investigations for this model to be optimal. This shows that even though the model performed well on the cross validation, it did not perform as well on the test set. This could be due to overfitting, and it is important to consider techniques to mitigate the overfitting.

Cost Sensitive Classifier Combined with Bagging and Decision Stump

Now we are going to create a cost sensitive classifier, combined with bagging and decision stump learners, with the number of iterations set to 25. We will try to find the optimal type 2 error for this model.

For cross validation, I found the optimal type 2 error to be at 6. Here are the details of this model:

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0.915	0.185	0.614	0.915	0.735	0.652	0.889	0.684	fp
	0.815	0.085	0.967	0.815	0.885	0.652	0.889	0.941	nfp
Weighted Avg.	0.839	0.109	0.881	0.839	0.848	0.652	0.889	0.878	

=== Confusion Matrix ===

```
  a   b   <-- classified as
 43   4 |   a = fp
 27 119 |   b = nfp
```

Important Numbers:
Precision: 0.881
Type 1 error rate: 18.5%
Type 2 error rate: 8.5%

This model has a high precision which is good as well as a lower type 2 error rate. Though the type 1 error rate is higher. When the type 2 error was set to lower numbers, such as 4 or 3, the type 1 error rate was lower. Though in these models, the type 2 error rate was higher. That is why I chose this as the optimal because type 2 error rate is more costly than type 1.

Now we are going to validate this model on the test set.

Cost Sensitive Classifier Combined with Bagging and Decision Stump using Test Set

Here are the details of this model:

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class

	0.917	0.247	0.550	0.917	0.687	0.587	0.850	0.577	fp
	0.753	0.083	0.965	0.753	0.846	0.587	0.850	0.922	nfp
Weighted Avg.	0.794	0.124	0.862	0.794	0.807	0.587	0.850	0.837	

=== Confusion Matrix ===

```
  a  b   <-- classified as
 22  2 |  a = fp
 18 55 |  b = nfp
```

Important Numbers:
Precision: 0.862
Type 1 error rate: 24.7%
Type 2 error rate: 8.3%

In this model, the precision maintains at a high rate and the type 2 error rate actually dropped slightly. The type 1 error rate raised higher at almost 25%. The type 1 error rate is this models main weakness, because the other aspects of this model are very good.

Cost sensitive Classifier combined with boosting (AdaBoostM1) and J48, with 25 iterations

Now I am going to create a cost sensitive classifier combined with boosting and J48. Remember, for this model, the iterations will continue to stay at 25 iterations. I am going to shift the type 2 error to find the optimal model.

I found that the optimal type 2 error is set at 3 to get the best model. Here are the details of this model:

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0.766	0.082	0.750	0.766	0.758	0.679	0.887	0.663	fp
	0.918	0.234	0.924	0.918	0.921	0.679	0.896	0.954	nfp
Weighted Avg.	0.881	0.197	0.882	0.881	0.881	0.679	0.894	0.883	

=== Confusion Matrix ===

```
  a   b   <-- classified as
 36  11 |   a = fp
 12 134 |   b = nfp
```

Important Numbers:
Precision: 0.882
Type 1 error rate: 8.2%
Type 2 error rate: 23.4%

This Model has a high precision and a low type 1 error rate. This models weakness is the type 2 error rate, which is fairly high at 23.4%. This is not very good because the type 2 error is more costly. More strategies to mitigate this may help to create a better model.

Now we are going to evaluate this model using a test data set.

Cost sensitive classifier combined with boosting (AdaBoostM1) and J48 Using Test Set, with 25 iterations

Here are the details of this model:

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0.583	0.082	0.700	0.583	0.636	0.535	0.892	0.695	fp
	0.918	0.417	0.870	0.918	0.893	0.535	0.882	0.949	nfp
Weighted Avg.	0.835	0.334	0.828	0.835	0.830	0.535	0.885	0.886	

=== Confusion Matrix ===

```
  a  b   <-- classified as
 14 10 |  a = fp
  6 67 |  b = nfp
```

Important Numbers:
Precision: 0.828
Type 1 error rate: 8.2%
Type 2 error rate: 41.7%

In this model, there is still a higher precision which is good. Furthermore, there is a low type 1 error rate which is also good. Thoug h, there is a very high type 2 error rate of 41.7% which is not good. I would not recommend using this model because of this high error rate. The reason for the jump between the cross validation model and the test set model may be due to overfitting, and it may be helpful to use strategies to help mitigate this.

Cost Sensitive Classifier combined with Boosting (AdaBoostM1) and Decision Stump, with 25 iterations

Now I am going to create a cost sensitive classifier combined with boosting and a decision stump learner. The number of iterations will continue to stay at 25 iterations. Again, we will change the type 2 error to find the optimal model.

I found that the optimal model was found when the type 2 error was changed to 4. Here are the details of this model:

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0.915	0.178	0.623	0.915	0.741	0.660	0.904	0.748	fp
	0.822	0.085	0.968	0.822	0.889	0.660	0.904	0.954	nfp
Weighted Avg.	0.845	0.108	0.884	0.845	0.853	0.660	0.904	0.904	

=== Confusion Matrix ===

```
  a   b   <-- classified as
 43   4 |   a = fp
 26 120 |   b = nfp
```

Important Numbers:
Precision: 0.884
Type 1 error rate: 17.8%
Type 2 error rate: 8.5%

This model is good, with a high precision and a low type 2 error rate. The type 1 error rate is higher, but not too high. Now, we are going to validate this model using a test set.

Cost Sensitive Classifier combined with boosting (AdaBoostM1) and Decision Stump Using Test Set, using 25 iterations

Here are the details of this model:

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0.833	0.178	0.606	0.833	0.702	0.597	0.890	0.738	fp
	0.822	0.167	0.938	0.822	0.876	0.597	0.890	0.948	nfp
Weighted Avg.	0.825	0.169	0.855	0.825	0.833	0.597	0.890	0.896	

=== Confusion Matrix ===

```
  a  b   <-- classified as
 20  4 |  a = fp
 13 60 |  b = nfp
```

Important Numbers:
Precision: 0.855
Type 1 error rate: 17.8%
Type 2 error rate: 16.7%

This model is also very good. The validated model has a slightly lower precision and a higher type 2 error rate, though it is still a good model.

Comparisons and Conclusions

In our last assignment, we created a cost sensitive tree similar to the models we have created here. We are going to compare each model to this cost sensitive tree.

This Model was found opimtal when **type 2 error was set to 2.** Here are the details of the cost sensitive tree obtained in assignment 2:

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0.894	0.144	0.667	0.894	0.764	0.686	0.853	0.579	fp
	0.856	0.106	0.962	0.856	0.906	0.686	0.853	0.921	nfp
Weighted Avg.	0.865	0.116	0.890	0.865	0.871	0.686	0.853	0.838	

=== Confusion Matrix ===

```
  a   b   <-- classified as
 42   5 |   a = fp
 21 125 |   b = nfp
```

Important Numbers:
Precision: 0.89
Type 1 error rate: 14.4%
Type 2 error rate: 10.6%

Comparison of the 1st model (bagging and J48):

For the first model, the iterations are equal to 10.

Here are the important numbers for this model:

Important Numbers:

Precision: **0.887**
Type 1 error rate: 22/(22+124) = **15.1%**
Type 2 error rate: 5/(5+42) = **10.6%**

In comparison to the Asiignment 2 model, this model is actually very similar. They actually have the saem type 2 error rate. The Assingment 2 model has a slightly higher precision and a slightly lower type 1 error, which may prove that the Asiignment 2 model may be slightly better.

For the second version of this model, the number of iterations is equal to 25.

Here are the details of this model:

Important Numbers:
Precision: 0.897
Type 1 error rate: 14.4%
Type 2 error rate: 8.5%

Compared to the model in assignment 2, this model has a lower type 2 error rate, the same type 1 error rate, and a higher precision. These models are similar, but this new model exceeds the performance of the model from assignment 2. Having a lower type 2 error rate is a strength of this model, as well as keeping the type 1 error rate the same. Furthermore, the precision of this model is almost 0.9, which is very very good. Overall, this model performed better than the cost sensitive tree obtained in assignment 2.

Comparison of the 2nd model (bagging and decision stump):

Here for the first model, the iterations are equal to 10.

Here are the important numbers of this model:

Important Numbers:
Precision: 0.88
Type 1 error rate: 15.1%
Type 2 error rate: 12.8%

Compared to the cost sensitive tree obtained in assignment 2, this model performed slightly worse. This model has a lower precision of about 0.01, and a higher type 1 and type 2 error rates. Though the difference between these numbers is small, it still shows that the cost sensitive tree obtained from assignment 2 performed better.

Here for the second model, the iterations are equal to 25.

Here are the important numbers of this model:

Important Numbers:
Precision: 0.881
Type 1 error rate: 18.5%
Type 2 error rate: 8.5%

Compared to the cost sensitive tree obtained in the last assignment, this model has a slightly lower precision as well as a higher type 1 error rate. Though on the other hand, the type 2 error rate is actually lower than the type 2 error rate in the assignment 2 model. This shows that both of these models have their weaknesses and strengths, and it is hard to determine which one may be better off of this. Because the type 2 error is expensive, this model may turn out to be better. Though having a higher type 1 error rate is also not ideal, so it is important to compare before deciding.

Comparison of the 3rd model (boosting and J48):

Here for the first model, the number of iterations is equal to 10.

Here are the important numbers of this model:

Important Numbers:
Precision: 0.888
Type 1 error rate: 8.2%
Type 2 error rate: 21.3 %

Compared to the model obtained in assignment 2, this model has a slightly lower precision of 0.002. Furthermore, it has a lower type 1 error rate and a higher type 2 error rate. Overall, I would say that the model obtained from assignment 2 performed better because of the lower type 2 error rate and the slightly higher precision.

Here for the second model, the number of iterations is equal to 25.

Here are the important numbers of this model:

Important Numbers:
Precision: 0.882
Type 1 error rate: 8.2%
Type 2 error rate: 23.4%

Compared to the model from assignment 2, this model has a lower precision, a lower type 1 error rate, and a higher type 2 error rate. Overall, the cost sensitive tree obtained from assignment 2 performed better because of the lower type 2 error rate. Even though the type 1 error rate is lower, type 2 error rate is more expensive.

Comparison of the 4th model(boosting and decision stump):

Here for the first model, the number of iterations is equal to 10.

Here are the important numbers of this model:

Important Numbers:
Precision: 0.884
Type 1 error rate: 17.8%
Type 2 error rate: 8.5%

Compared to the cost sensitive tree obtained in assignment 2, this model has a lower precision, a higher type 1 error rate and a lower type 2 error rate. A strength of this model is the lower type 2 error rate. Though it has a slightly lower precision of 0.006 and a higher type 1 error rate, this model may be the better performer due to the lower type 2 error rate.

Here for the second model, the number of iterations is equal to 25.

Here are the important numbers for this model:

Important Numbers:
Precision: 0.884
Type 1 error rate: 17.8%
Type 2 error rate: 8.5%

This model turned out to stay the same regardless of the number of iterations, Therefore the conclusion and comparison will also be the same. These models may end up being the better choice due to the lower type 2 error rates, even though there is a slightly lower precision and a higher type 1 error rate. Both models are good performers though and both gave good results.